CLAIMS

I claim:

1.   An electronic voting system for use with a computerized network, comprising:

a plurality of voting computers coupled to the computerized network, wherein each voting computer provides an electronic encrypted ballot, wherein each electronic ballot is encrypted under a discrete log asymmetric encryption process using underlying groups Z_p or elliptic curve;

at least first, second and third authority computers coupled to the computerized network, wherein the first authority computer is configured to receive a series of electronic ballots corresponding to an aggregation of each of the electronic ballots received from the plurality of voting computers, and to apply a secret, one-way cryptographic transformation using at least a first secret key to anonymously shuffle the series of electronic ballots and produce a first shuffled series of ballots, wherein only the first authority computer knows a correspondence between the first series of shuffled ballots and the series of electronic ballots, and wherein the first authority computer is further configured to provide a first linear size, non-interactive proof of correctness for the first series of shuffled ballots based on a scaled iterated logarithmic multiplication proof;

wherein the second authority computer is configured to receive the first series of shuffled ballots, to apply the cryptographic transformation using at least a second secret key to anonymously shuffle the first series of shuffled ballots and produce a second series of shuffled ballots, wherein only the second authority computer knows a correspondence between the first series of shuffled ballots and the second series of shuffled ballots, and wherein the second authority computer is

26 further configured to provide a second linear size, non-interactive proof of
27 correctness for the second series of shuffled ballots based on the scaled
28 iterated logarithmic multiplication proof;

29 wherein the third authority computer is configured to receive
30 the second series of shuffled ballots, to apply the cryptographic
31 transformation using at least a third secret key to anonymously shuffle the
32 second series of shuffled ballots and produce a third series of shuffled
33 ballots, wherein only the third authority computer knows a correspondence
34 between the third series of shuffled ballots and the second series of shuffled
35 ballots, and wherein the third authority computer is further configured to
36 provide a third linear size, non-interactive proof of correctness for the third
37 series of shuffled ballots based on the scaled iterated logarithmic
38 multiplication proof; and

39 a verification computer coupled to the computerized network,
40 wherein the verification computer is configured to receive the proofs of
41 correctness from the first, second and third authority computers and without
42 interacting with the first, second and third authority computers, to verify a
43 correctness of the shuffled ballots.

1 2. The system of claim 1, further comprising:

2 a server computer system coupled to the computerized
3 network, wherein the server computer system is configured to: receive the
4 plurality of electronic ballots from the plurality of voting computers; verify
5 a proof of validity of each of the plurality of electronic ballots; form an
6 encrypted tally of the votes from the plurality of electronic ballots; transmit
7 the encrypted tally to the first, second and third authority computers;
8 receive ballot decryption shares produced from at least two of the first,
9 second and third authority computers; and compute a decrypted tally; and

10   at least one voting poll computer coupled to the computerized

11 network and providing some of the plurality of electronic encrypted ballots

12 to the server computer system.

1   3.   The system of claim 1 wherein the first, second and

2 third authority computers are configured to provide Chaum-Pedersen proofs

3 for the first, second and third shuffles of the ballots, respectively, and

4 wherein each of the first, second and third authority computers generate an

5 initial challenge series, receive a challenge from at least one verification

6 computer, and generate the cryptographic transformation based on an

7 exponentiation of the initial and received challenges.

1   4.   The system of claim 1 wherein the computerized

2 network includes the World Wide Web, wherein each of the plurality of

3 voting computers and first, second and third authority computers include a

4 web browser program.

1   5.   The system of claim 1 wherein the plurality of voter

2 computers include at least one palm-sized computer, cell phone, wearable

3 computer, interactive television terminal or Internet appliance.

1   6.   A computer system for receiving a sequence of

2 elements, comprising:

3   a server computer coupled to a computer network and

4 configured to:

5   receive a sequence of electronic data elements

6 representing individual data files,

7   apply a cryptographic transformation using at least a

8 first secret key to anonymously permute the sequence of electronic

9 data elements and produce a first shuffled sequence of electronic

10   data elements, wherein the server computer knows a correspondence

11   between the first shuffled sequence of electronic data elements and

12   the sequence of electronic data elements, and

13   generate a first linear size proof of correctness for the

14   first shuffled sequence of electronic data elements based on a scaled

15   iterated logarithmic multiplication proof.

1   7.   The system of claim 6 wherein the received sequence

2   of electronic data elements are encrypted using Z_p or elliptic curve groups

3   using a key unknown to the server computer, and wherein the server

4   computer is further configured to:

5   receive a series of randomly generated values e_i from a

6   verifier computer;

7   secretly generate a series of values U_i based on a secret, one-

8   way cryptographic transformation that employs the received series of values

9   e_i and secretly generated values

$$\overline{u_i}$$

10   permute the sequence of electronic data elements to produce

11   the first shuffled sequence of elements based on the series of values U_i and a

12   secret value d; and

13   provide the values U_i and a series of proof values based on the

14   cryptographic transformation as a proof of knowledge that the server

15   computer has access to how the cryptographic transformation permuted the

16   sequence of electronic data elements to produce the first shuffled sequence

17   of elements without revealing the cryptographic transformation to the

18   verifier computer.

1   8.   The system of claim 6 wherein the server computer is

2   further configured for:

3   receiving a plurality of public keys from a corresponding

4   plurality of individuals, wherein each of the plurality of individuals have a

5   private key corresponding to one of the plurality of public keys;

6   receiving a request for a certificate from one of the plurality

7   of individuals having a one private key;

8   providing at least a subset of the plurality of public keys to the

9   requesting individual;

10   receiving a shuffle of the plurality of public keys and a linear

11   size proof of correctness for the shuffled public keys based on a scaled

12   iterated logarithmic multiplication proof and a value corresponding to the

13   one private key, wherein the value provides proof that the one individual

14   has knowledge of the one private key without revealing the one private key;

15   checking the proof of correctness;

16   checking that the value is mathematically related to a one of

17   the public keys that corresponds to the one private key;

18   issuing a certificate to the one individual; and

19   reducing the plurality of public keys by the one public key.

1   9.   The system of claim 6 wherein the sequence of

2   electronic elements are public keys, and wherein the server if further

3   configured to check, in response to a request from an individual, that the

4   individual has a value uniquely and mathematically related to a one of the

5   public keys; and

6   if so, issue a certificate to the one individual.

1   10.   A computer-implemented method, comprising:

2   receiving a plurality of public keys from a corresponding

3   plurality of individuals, wherein each of the plurality of individuals have a

4   private key corresponding to one of the plurality of public keys;

5   receiving a request for a certificate from one of the plurality

6   of individuals having a one private key;

7   providing at least a subset of the plurality of public keys to the

8   requesting individual;

9   receiving a shuffle of the plurality of public keys and a linear

10   size proof of correctness for the shuffled public keys based on a scaled

11   iterated logarithmic multiplication proof and a value corresponding to the

12   one private key, wherein the value provides proof that the one individual

13   has knowledge of the one private key without revealing the one private key;

14   checking the proof of correctness;

15   checking that the value is mathematically related to a one of

16   the public keys that corresponds to the one private key;

17   issuing a certificate to the one individual; and

18   reducing the plurality of public keys by the one public key.

1   11.   The method of claim 10 wherein the method further

2   includes setting a value G to a subgroup operator g from an Z_p or elliptic

3   curve group, wherein providing at least a subset of the plurality of public

4   keys includes providing all of the then current public keys H.

1   12.   The method of claim 10 wherein providing at least a

2   subset of the plurality of public keys includes providing at least a subset of

3   a plurality of public key pairs, wherein receiving a shuffle of the plurality of

4   public keys includes receiving a shuffle of a true subset of the plurality of

5   public key pairs as selected by the one individual.

1   13.   The method of claim 10, further comprising:

2   receiving from each of a plurality of authorities, in sequence,

3   a shuffled set of the plurality of public keys H' based on a secret

4   cryptographic shuffle operation performed on at least a subset of the

5   plurality of public keys to produce the shuffled set of the plurality of public

6   keys H';

7   receiving from each of a plurality of authorities, in sequence,

8   a verification transcript of the cryptographic shuffle operation; and

9   verifying a correctness of the cryptographic shuffle operation

10   based on the verification transcript; and if verified, then setting at least a

11   subset of the plurality of public keys to H to H'.

1   14.   The method of claim 10 , further comprising:

2   at a time after receiving at least some of the plurality of public

3   keys, setting at least a subset of the then received plurality of public keys to

4   a received shuffled set of the plurality of public keys, wherein the shuffled

5   set of the plurality of public keys have been received from a third party.

1   15.   The method of claim 10, further comprising:

2   receiving the issued certificate from the one of the plurality of

3   individuals; and

4   providing an electronic ballot to the one individual.

1   16.   The method of claim 10 wherein issuing a certificate

2   includes digitally signing the received request to produce a public key

3   infrastructure ("PKI") certificate.

1   17.   The method of claim 10 , further comprising:

2   receiving issued certificates from at least some of the plurality

3   of individuals and providing initial electronic ballots in response thereto;

4   and

5   receiving unencrypted voted ballots from the at least some of

6   the plurality of individuals.

1   18.   A   computer-implemented   cryptographic   method

2   between a prover computer and a verifier computer, the method comprising:

3   selecting a subgroup generator g selected from a group G;

4   secretly generating a prover key c, and a commitment value C

5   based on the subgroup generator g;

6   secretly establishing a cryptographic relationship between first

7   and second sequences of elements;

8   providing to the verifier computer the commitment C and the

9   first and second sequences of elements, but not the cryptographic

10   relationship;

11   computing a series of proof values based on the cryptographic

12   relationship; and

13   providing the series of computed proof values to the verifier

14   computer as a non-interactive proof of knowledge that the prover computer

15   has access to the cryptographic relationship without revealing the

16   cryptographic relationship to the verifier computer.

1   19.   The method of claim 18 wherein at least the second

2   sequence of elements is a sequence of encrypted ballots, wherein each

3   ballot is encrypted using Z_p or elliptic curve groups;

4   wherein the first and second sequences of elements are

5   respectively

$$(X_1,...,X_k) \quad and \quad (Y_1,...,Y_k)$$

6   wherein the first and second sequence of elements have the

7   cryptographic relationship

8   and wherein computing and providing the series of proof

$$(g^{u_1}, \ldots, g^{u_k}) = (X_1, \ldots, X_k)$$

$$(g^{v_1}, \ldots, g^{v_k}) = (Y_1, \ldots, Y_k) \text{ and where}$$

$$c^k \prod_{i=1}^{k} u_i = \prod_{i=1}^{k} v_i$$

9   values includes providing Chaum-Pedersen proofs based on:

for each $0 \leq i \leq k$ generate random r_i

$$R_i = g^{r_i}$$

for each $1 \leq i \leq k$ $w_i = r_i u_i / r_{i-1}$

$$W_i = g^{w_i}$$

$$z_i = w_i / v_i$$

$$Z_i = g^{z_i}$$

10

11   wherein the Chaum-Pedersen proofs provided to the verifier computer are

12   of a form:

$$(R_{i-1}, X_i, R_i, W_i) \text{ and } (Y_i, C, W_i, Z_i).$$

13

1   20.   The method of claim 18, further comprising:

2   permuting the first sequence of elements to produce the

3   second sequence of elements based on a cryptographic transformation;

4   receiving a randomly generated value t from the verifier

5   computer;

6   secretly generating a value T based on the received value t and

7   the subgroup generator, and secretly generating a value S based on the

8   received value t and the prover key c; and

9 wherein computing and providing to the verifier computer the

10 series of proof values includes providing a series of values based on the

11 cryptographic transformation as a proof of knowledge that the prover

12 computer has access to how the cryptographic transformation permuted the

13 first sequence of elements to produce the second sequence of elements

14 without revealing the cryptographic transformation to the verifier computer.

1 21. The method of claim 18, further comprising:

2 permuting the first sequence of elements to produce the

3 second sequence of elements based on a cryptographic transformation in a

4 form of

$$(g^{u_1},\ldots,g^{u_k}) = (X_1,\ldots,X_k)$$
$$(g^{cu_{\pi(1)}},\ldots,g^{cu_{\pi(k)}}) = (Y_1,\ldots,Y_k)$$

5 receiving a randomly generated value t from the verifier

6 computer;

7 secretly generating a value T based on raising the subgroup

8 generator g to the received value t, and secretly generating a value S based

9 on raising the value T to the prover key c; and

10 wherein computing and providing to the verifier computer the

11 series of proof values includes providing a series of values based on the

12 cryptographic transformation in a form of:

$$U_i = X_i / T$$
$$V_i = Y_i / S$$

13 as a proof of knowledge that the prover computer has access to how the

14 cryptographic transformation permuted the first sequence of element to

15 provide the second sequence of elements without revealing the

16 cryptographic transformation to the verifier computer.

1   22.   The method of claim 18, further comprising:

2   receiving the first sequence of elements as a set of elements

3   that have previously been permuted in a manner unknown to the prover

4   computer;

5   receiving a series of randomly generated values e_i from the

6   verifier computer;

7   secretly generating a series of values U_i based on a secret

8   cryptographic transformation that employs the received series of values e_i

9   and secretly generated values

     $\overline{u}_i$

10   permuting the second sequence of elements with respect to the

11   first sequence of elements based on the series of values U_i and a secret value

12   d; and

13   wherein computing and providing to the verifier computer the

14   series of proof values includes providing the resulting values U_i and

15   providing a series of proof values based on the cryptographic transformation

16   as a proof of knowledge that the prover computer has access to how the

17   cryptographic transformation permuted the first sequence of element to

18   provide the second sequence of elements without revealing the

19   cryptographic transformation to the verifier computer.

1   23.   The method of claim 18, further comprising:

2   receiving the first sequence of elements as a set of elements

3   that have previously been permuted in a manner unknown to the prover

4   computer;

5   receiving a series of randomly generated values e_i from the

6   verifier computer;

7   secretly generating a series of values U_i based on a secret

8   cryptographic transformation of a form

$$u_i = \overline{u}_i + e_i = \log_g U_i$$

9

10   permuting the second sequence of elements with respect to the

11   first sequence of elements based on the series of values U_i and a secret value

12   d based on the following operations

$$(V_1,\ldots,V_k) = (U_{\pi(1)}^d,\ldots,U_{\pi(k)}^d)$$
$$D = g^d$$
$$v_i = \log_g V_i$$
$$A_i = X_i^{v_i}$$
$$B_i = Y_i^{u_i}$$

13

14   and wherein computing and providing to the verifier the series of proof

15   values includes providing the resulting values $U_{i,}$,

$$A = \prod_{i=1}^{k} A_i$$

$$B = \prod_{i=1}^{k} B_i$$

16   and for $1 \leq i \geq k$, providing a series of proof Chaum-Pedersen of a form

17

$$(g,V_i,X_i,A_i) \quad \text{and} \quad (g,U_i,Y_i,B_i)$$

18   and a Chaum-Pedersen proof for (D, A, C, B) as a proof of knowledge that

19   the prover computer has access to how the cryptographic transformation

20   permuted the first sequence of element to provide the second sequence of

21   elements without revealing the cryptographic transformation to the verifier

22   computer.

1   24.   The method of claim 23, further comprising repeating

2   the receiving the first sequence of elements, receiving a series of randomly

3   generated values, secretly generating a series of values, and permuting the

4 second sequence of elements for *l*-tuple of elements in the first sequence of

5 elements.

1 25. The method of claim 22 wherein receiving the first

2 sequence of elements includes receiving a subset of a set of identifying

3 elements, wherein each identifying element in the set corresponds to an

4 individual, and wherein the method further comprises:

5 receiving an anonymous certificate if the verifying computer

6 verifies the series of proofs.

1 26. The method of claim 18 wherein the group G is Z_p.

1 27. The method of claim 18 wherein the group G is an

2 elliptic curve group.

1 28. A computer-readable medium whose contents provide

2 instructions, when implemented by a computer, perform a shuffling of a

3 sequence of electronic data elements, comprising:

4 receive the sequence of electronic data elements;

5 apply a secret, one-way cryptographic transformation using at

6 least a first secret key to anonymously permute the sequence of electronic

7 data elements and produce a first shuffled sequence of electronic data

8 elements; and

9 generate a first linear size, non-interactive proof of

10 correctness for the first shuffled sequence of electronic data elements based

11 on a scaled iterated logarithmic multiplication proof.

1   29.   The computer-readable medium of claim 28 wherein the received

2 sequence of electronic data elements are encrypted with an underlying mathematical group

3 being a ring of integers having a modulus integer value p (Z_p).

1   30.   The computer-readable medium of claim 28 wherein the

2 computer-readable medium is a logical node in a computer network receiving

3 the sequence of electronic data elements and the contents.

1   31.   The computer-readable medium of claim 28 wherein the

2 computer-readable medium is a computer-readable disk.

1   32.   The computer-readable medium of claim 28 wherein the

2 computer-readable medium is a data transmission medium transmitting a

3 generated data signal containing the contents.

1   33.   The computer-readable medium of claim 28 wherein

2   the computer-readable medium is a memory of a computer system.

1   34.   The computer-readable medium of claim 28 wherein

2   the computer-readable medium is an Internet connection link to a voting

3   authority server computer.

1   35.   In a cryptographic method, a transmitted signal for use by a computer,

2 comprising:

3   a shuffled sequence of electronic data elements representing individual data

4 files, wherein a one-way cryptographic transformation using at least a first secret key

5 anonymously permuted an input sequence of electronic data elements to produce the shuffled

6 sequence of electronic data elements, and

7   a linear size proof of correctness for the shuffled sequence of electronic data

8 elements based on a scaled iterated logarithmic multiplication proof.